PATENT

AF/2155
/120

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

#30
LDJ
9-3-04

Applicant(s):	Marc Tremblay and William Joy		
Title:	LOCAL AND GLOBAL REGISTER PARTITIONING IN A VLIW PROCESSOR		
Application No.:	09/204,585	Filed:	December 3, 1998
Examiner:	David Y. Eng	Group Art Unit:	2155
Atty. Docket No.:	004-3288		

August 20, 2004

Mail Stop Appeal Brief - Patents
COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, VA 22313

APPELLANT'S BRIEF (37 C.F.R. § 1.192)

This brief is in furtherance of the Notice of Appeal, filed on February 20, 2004. The fees required under § 1.17(c), are provided in the accompanying Transmittal. The Notice of Appeal was received by the Office on February 20, 2004 and a Request for Extension of Time is filed herewith, thereby setting August 20, 2004 as the date for filing Appellant's Brief. This brief is being transmitted in triplicate pursuant to 37 C.F.R. § 1.192(a).

REAL PARTY IN INTEREST

The real party in interest in this appeal is Sun Microsystems, Inc., as evidenced by the assignment recorded at Reel 9619/Frame 0633.

RELATED APPEALS AND INTERFERENCES

Appellants have no knowledge of any related appeals or interferences.

STATUS OF CLAIMS

Claims 1-28 are presented herein on appeal. Claim 29 was cancelled by the Applicants in response to a non-final Office action dated February 25, 2002. Claims 1, 7-9, 11-15, 20-23, and

27-28 have been amended during prosecution, while claims 2-6, 10, 16-19, and 24-26 remain as originally filed. Most recently, Claims 1, 15, and 23 were amended on September 9, 2003, in response to a non-final Office action dated April 9, 2003. Claims 1-28, as presented on September 9, 2003, were rejected in a final Office action dated November 25, 2003. That final rejection is now appealed.

Claims 1-28, now presented herein on appeal, are reproduced in the Appendix attached hereto.

STATUS OF AMENDMENTS

No amendments after final rejection have been submitted.

SUMMARY OF INVENTION

This presently claimed invention relates to processors storing information in register files, and more particularly to processors employing register files divided into multiple register file segments each coupled to and associated with a respective functional unit, wherein each register file segment is partitioned (or partitionable) into local and global registers. The register file and/or register file segments thereof are implemented as an addressable array that is partitionable into global and local register portions. In some embodiments, such a partition is programmable.

Referring to Fig. 6, for example, a schematic block diagram shows a register file 600 for an exemplary very long instruction word (VLIW) processor 100 that includes an implementation of global and local register partitioning. The VLIW processor has a plurality of functional units including three media functional units 622, 624, and 626, and a general functional unit 620. The processor 100 also includes a multi-port register file 600 that is divided into a plurality of separate register file segments 610, 612, 614, and 616, each of the register file segments being associated to one of the plurality of functional units. The register file segments 610, 612, 614, and 616 are partitioned into local registers and global registers. The global registers are read and written by all functional units 620, 622, 624, and 626. The local registers are read and written only by a functional unit associated with a particular register file segment. The local registers and global registers may be addressed using register addresses in an address space that is

separately defined for a register file segment/functional unit pair including register file segment 610/general functional unit 620, register file segment 612/media functional unit 622, register file segment 614/media functional unit 624, and register file segment 616/media functional unit 626.

In one embodiment, the global registers are addressed within a selected global register range using the same register addresses for the plurality of register file segment/functional unit pairs, for example, global registers 0-95. The local registers in a register file segment are addressed using register addresses in a local register range outside the global register range, for example addresses 96-127, that are assigned within a single register file segment/functional unit pair. Register addresses 96-127 applied to the register file segments in the local register range are the same for the plurality of register file segment/functional unit pairs and address registers locally within a register file segment/functional unit pair. The register specifiers of the local registers, as defined external to the processor (e.g., by a compiler), do not overlap, but instead have distinct and different specifiers. For example, in one embodiment, 96 global registers are addressed using address specifiers 0-95 in all of the four register file segments. Local registers 96-127 in the register file segment 610, local registers 128-159 in register file segment 612, local registers 160-191 in register file segment 614, and local registers 192-223 in register file segment 616 are all addressed using register addresses 96-127. In this example, the total number of distinct and independent registers is 96+(4*32) = 224. The 224 registers are addressed using 7 bits that define an address space from 0-127, rather than the 8 bits that are otherwise required to access 224 registers. In a general case, the number of registers can be expressed as $N_G + (M*N_L) = N$, with the number of bits being used to address N registers being equal to the number of bits B that are used to address $N=2^B$.

In some VLIW exploitations, global and local register partitioning advantageously leverages the information content of register specifier bits in an instruction word by inherently communicating information by position dependence within a VLIW instruction group. The positioning of a register specifier in the instruction word thus communicates addressing information. The additional information allows a compiler or programmer to specify more registers in fewer bits than have been specified conventionally. One address bit is thus saved for each of the four subinstruction positions, a savings of four bits per subinstruction and a savings of 16 bits per VLIW instruction. The reduction in address bits is highly advantageous in a VLIW

processor that includes powerful functional units that execute a large plurality of instructions, each of which is to be encoded in the VLIW instruction word.

In general embodiments, the register file 600 includes N physical registers. The N-register register file 600 is duplicated into M register file segments 610, 612, 614, and 616, each having a reduced number of read and/or write ports in comparison to a nonduplicated register file, but each having the same number of physical registers. The register file segments are partitioned into N_G global and N_L local register files where N_G plus N_L is equal to N. The register file operates equivalently to having N_G +(M* N_L) total registers available for the M functional units. The number of address bits for addressing the N_G +(M* N_L) total registers remains equal to the number of bits B that are used to address $N=2^B$ registers. The local registers for each of the M register file segments are addressed using the same B-bit values.

In some embodiments, partitioning of the register file 600 is programmable so that the number N_G of global registers and number N_L of local registers is selectable and variable. For example, a register file including four register file segments each having 128 registers may be programmably configured as a flat register file with 128 global registers and 0 local registers with the 128 registers addressed using seven address bits. Alternatively, the four register file segments may be programmably configured, for example, to include 64 global registers and 64 local registers so that the total number of registers is 64+(4*64)=320 registers that are again addressed using 7 bits rather than the 9 bits that would otherwise be required to address 320 registers.

Suitable array configurations, decode logic, read and write port configurations, word and bit line configurations and other particulars for exemplary implementations of a register file (and individual segments thereof) are described in the specification in greater detail with reference to FIGS. 7, 8A, 8B and 9.

ISSUES

1. Properly interpreted, neither *Yung* nor *Luan* discloses or suggests register file segments each coupled to and associated with a respective functional unit and each implemented as an addressable array and partitionable into global and local registers.

2. No proper construction of *Luan* is consistent with an interpretation of *Luan's* configurable memory bank allocation technique as register file segments that are partitionable into global registers and local registers. Memory is not registers and *Luan's* exploitation of selective allocation of memory banks is neither analogous to Applicant's programmable partitioning of a register file segment implemented as an addressable array, nor suitable for use in combination with *Yung*.

GROUPING OF CLAIMS

Group 1: 1-7, 9-14 and 23-27;

Group 2: 15-20 and 22 (claims do not stand or fall together); and

Group 3: 8, 21, 28 (claims do not stand or fall together).

ARGUMENTS

Despite express claim language that recites register file segments ***each coupled to and associated with a respective functional unit*** and each "***implemented as an addressable array and partitionable into global and local registers***," examination remains mired in a mischaracterization of U.S. Patent 5,592,679 to Yung. The *Yung* does not disclose or suggest an addressable array that is partitionable, but rather implements a hierarchical register architecture in which a fixed global register file is shared amongst functional units and fixed and separate local buffers are defined for respective functional units to cache register values close to respective uses thereof.

No proper construction of Yung's disclosure is consistent with an interpretation of Yung's entirely separate global registers and local buffers as a ***single addressable array that is partitionable*** *into global and local portions.*

Furthermore, in some independent claims, Applicant positively recites limitations related to ***programmably configuring*** the partition to vary relative numbers of registers of the ***addressable array*** configured as global or local. The Office further relies on U.S. Patent 5,911,149 to Luan et al. *Luan* concerns a computer system in which a shared memory subsystem

thereof is programmable to selectively define banks of the memory as either dedicated to the processor or shared by the processor and any peripherals. *Luan* operates by selectively coupling respective memory banks to an appropriate one of two busses (i.e., to a first bus connected to the processor or to a second one coupled to peripherals and to the processor).

No proper construction of Luan's disclosure is consistent with an interpretation of Luan's configurable memory bank allocation technique as register file segments that are partitionable into global registers and local registers. Memory is not registers and Luan's exploitation of selective allocation of memory banks is neither analogous to Applicant's programmable partitioning of a register file segment implemented as an addressable array nor suitable for use in combination with Yung.

The Office rejects under 35 USC § 103 and relies on *Luan* or *Yung* in view of *Luan* for each and every rejection. Specific claimed limitations are not disclosed or suggested in either *Luan* or *Yung*. As a result, the prior art has been accorded inordinate scope, the claims have been interpreted in a way that discounts claim limitations, or both. Careful analysis will show that claim limitations are neither expressly nor inherently disclosed in any of the relied upon references. As a result, no *prima facie* case of obviousness has been made out. The nature of this legal error is now summarized.

Obviousness Rejections under 35 U.S.C. § 103

Claims 1, 3-14 and 23-28 all stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent 5,592,679 to Yung (*Yung*) in view of U.S. Patent 5,911,149 to Luan et al (*Luan*). Claims 1, 3-14 and 23-28 are further rejected under 35 U.S.C. § 103(a) as being unpatentable over *Luan*. Claims 2 and 15-22 all stand rejected under 35 U.S.C. § 103(a) as being unpatentable over *Yung* and *Luan* further in view of U.S. Patent 6,023,575 to Nishimoto (*Nishimoto*). Claims 1, 3-14 and 23-28 are further rejected under 35 U.S.C. § 103(a) as being unpatentable over *Luan* in view of *Nishimoto* .

The Office relies on *Nishimoto* solely for disclosure of a VLIW processor and for no other reason. Without addressing whether *Nishimoto* is properly combinable with the other references, but rather to simplify issues for appeal, Applicant does not argue that mere recitation

that a processor is a VLIW processor or feature thereof constitutes a patentable distinction. Accordingly, disposition of this Appeal depends on proper interpretation of *Yung* and *Luan*.

The legal standard for obviousness is defined in the Patent Statute, 35 U.S.C. § 103, which specifies, in addition to novelty requirements under § 102, further conditions for patentability relating to nonobvious subject matter. Those further conditions include the following:

> [a] patent may not be obtained though the invention is not identically disclosed or described [by prior art under 35 U.S.C. § 102] if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains.

35 U.S.C. § 103 (1999).

Obviousness is a legal determination based on underlying factual inquiries. *Minnesota Min. & Mfg. Co. v. Johnson & Johnson Orthopedics, Inc.*, 24 USPQ2d 1321, 1332-1333 (Fed. Cir. 1992). *Graham v. John Deere Co.*, 383 U.S. 1, 17 (1966) defines the factual inquiries utilized to evaluate the prior art. Specifically, the prior art is evaluated in terms of: (1) its scope and content; (2) the differences between the prior art and the claimed invention; (3) the level of ordinary skill in the art at the time the application was filed; and (4) objective, or secondary, evidence of nonobviousness such as commercial success, failure of others, long-felt need and unexpected results, which must be considered in reaching a conclusion of obviousness. *Graham v. John Deere Co.*, 383 U.S. 1, 17, 148 U.S.P.Q. 459, 460 (1966); *Panduit Corp. v. Dennison Mfg. Co.*, 810 F.2d 1561, 1566-67, 1 U.S.P.Q.2d 1593, 1595-96 (Fed. Cir. 1987); *Minnesota Min. & Mfg. Co. v. Johnson & Johnson Orthopaedics, Inc.*, 24 U.S.P.Q.2d 1321, 1333 (Fed. Cir. 1992). In the present appeal, pertinent issues relate primarily to specific differences between the prior art and appealed claims, specifically absence, in the relied upon references, of certain features of the appealed claims.

Obviousness analysis begins with a key legal question—what is the invention claimed? In this regard, the claimed invention must be evaluated as a whole. 35 U.S.C. § 103; *see also Panduit Corp.*, 1 U.S.P.Q.2d at 1597. Fundamentally, all claim limitations must be considered in

the obviousness analysis. Indeed, it is clear error to ignore limitations clearly set forth in the claims. *Panduit Corp.*, 1 U.S.P.Q.2d at 1604. In general, multiple prior art references may be combined to provide a basis for an obviousness determination; however, there must be some teaching or suggestion for the combination. *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1456 (Fed. Cir. 1998). Finally, a prior art reference must be considered in its entirety, *i.e.*, as a *whole*, including portions that would lead away from the claimed invention. *W.L. Gore & Associates, Inc. v. Garlock, Inc.*, 220 U.S.P.Q. 303, 311 (Fed. Cir. 1983). Indeed, it is impermissible within the framework of § 103 to pick and choose from any one reference only so much of it as will support a given position, to the exclusion of other parts necessary to the full appreciation of what such reference fairly suggests to one of ordinary skill in the art. *Bausch & Lomb, Inc. v. Barnes-Hind, Inc.*, 230 U.S.P.Q. 416 (Fed. Cir. 1986); *see also In re Wright*, 9 U.S.P.Q.2d 1649, 1652 (Fed. Cir. 1989).

Group 1 (claims 1-7, 9-14 and 23-27)

Claims 1-7, 9-14 and 23-27 all stand rejected under 35 U.S.C. § 103(a) as being unpatentable over *Yung* in view of *Luan* or simply over *Luan*.[1]

Applicant respectfully notes that neither *Yung* nor *Luan* discloses or suggests ***register file segments each coupled to, and associated with, a respective functional unit*** and each ***implemented as an addressable array and partitionable into global and local registers***. In particular, *Yung* does not disclose or suggest a register file segment that is implemented as an addressable array and that is partitionable, but rather implements a hierarchical register architecture in which a fixed global register file (290) is shared amongst functional units and in which fixed and separate local buffers (241d, 242d ...) are defined for respective functional units to cache register values close to respective uses thereof. *See Yung*, Fig. 2, col. 6, lines 18-44. Global register file (290) and a separate local buffer (241d or 242d) do not together constitute a "register file segment" as claimed and cannot be said to be associated with any respective functional unit. Indeed, no association exists between any global register and any particular functional unit in *Yung*.

[1] Strictly speaking, the Office's rejection of claim 2 relies further on *Nishimoto*. As previously discussed, *Nishimoto* does not affect outcome of this Appeal since Applicants do not argue that mere recitation that a processor

Neither *Luan* nor *Nishimoto* provides the missing disclosure. Accordingly, for at least this reason, all rejections (based on *Yung*) of the group 1 claims are unsustainable and should be reversed.

Furthermore, whatever structure(s) in *Yung* the Office chooses to rely upon to constitute a "`register file segment`" coupled to, and associated with, a respective functional unit, the number of global registers and the number of local registers is ***not programmably configurable*** as variously recited in the independent claims of group 1. For this reason as well, rejections (based on *Yung*) of the group 1 claims are unsustainable. The Office has relied upon *Luan* for disclosure of a programmably configurable partition. However, as described more completely below, such reliance is inapposite.

The Office rejects group 1 claims based on *Luan* apart from *Yung*, accordingly Applicants first address *Luan* and then combination of *Yung* and *Luan*.

Luan is also devoid of disclosure of either (i) a register file segment that is implemented as an addressable array and that is partitionable or (ii) a programmably configurable partition of a register file segment. To the contrary, *Luan* discloses a:

> `programmable shared memory system and method [that] selectively dedicates a first portion of memory to use by a processor and allocates a second portion of memory to shared use by the processor and any peripherals in the system. The programmable memory architecture is implemented using a dual bus architecture and a plurality of configurable memory banks. The dual bus architecture has a first bus connected to the processor and a second bus coupled to the peripherals by peripheral controllers and also coupled to the processor by a system controller.`

Luan, col. 2, lines 3-13.

To even attempt to interpret *Luan's* programmable shared memory as the same as (or interchangeable with) a register file segment that is implemented as an addressable array and that is partitionable, simply neglects the clearly distinct roles memory and registers play in computer system architecture. Computer systems consistent with Applicant's design, *Yung's* design or *Luan's* design, each demonstrably or necessarily and inherently employ both memory and

---

is a VLIW processor constitutes a patentable distinction. Applicant's arguments stand or fall based on proper interpretation of *Yung* and *Luan*.

registers. That memory and those registers are separate and distinct structures; they are employed for different purposes and they are not the same or interchangeable. Therefore, whatever *Luan's* teaching with respect to selectively coupling memory banks to an appropriate bus to define first and second portions of a memory subsystem, *Luan* does not disclose or suggest a register file segment that is implemented as an addressable array and that is programmably partitionable. For at least this reason, all rejections (based on *Luan*) of the group 1 claims are unsustainable and should be reversed.

However, even beyond the substantial disconnect between *Luan's* techniques Applicant's claimed register configurations, in any attempt to map *Luan's* structures onto Applicant's claims, the Office must construe a group (i.e., two or more) of *Luan's* memory banks to be a register file segment coupled to and associated with a respective functional unit and partitionable into global registers and local registers. No reasonable interpretation of *Luan* supports an association between any group of *Luan's* memory banks and any particular functional unit of *Luan's* processor. For at least the foregoing reasons, the Office's rejection of claims of group 1 over *Luan* is unsustainable and should be reversed.

Turning now to rejections based on *Yung* in view of *Luan*, Applicant respectfully notes that (as demonstrated above) neither *Yung* nor *Luan* discloses or suggests register file segments each coupled to, and associated with, a respective functional unit and each implemented as an addressable array and partitionable into global and local registers. Accordingly, no *prima facie* case of obviousness has been made out. Furthermore, any combination of *Yung* nor *Luan* necessarily rests on the premise that programmable partitionability of *Luan's* memory subsystem may be imported into *Yung* to somehow provide programmable partitionability of storage provided by *Yung's* local buffers and global register file. As previously explained, registers and memory are not the same or interchangeable as employed by Applicant, *Yung* or *Luan*. Accordingly, whatever *Luan's* teaching with respect to memory banks, it is simply not applicable to registers. However, even more important, there is simply no way that *Luan's* techniques could be employed to cause an entry of *Yung's* local buffers to behave as a global register or to cause an entry of *Yung's* global register file to behave as a local buffer. Therefore, no programmability of register file partition may be added to *Yung* simply by applying *Luan's* techniques.

Accordingly, for at least the foregoing reasons, the Office's rejection of claims of group 1 over *Yung* in view of *Luan* are not sustainable and should be reversed.

Action Requested: For at least the reasons set forth above, the references fail to teach or suggest, alone or in combination, all elements of independent claims 1 and 23. Accordingly, all claims of Group 1 (claims 1-7, 9-14 and 23-27) as well as those dependent therefrom (claims 8 and 28) are allowable and rejections under 35 U.S.C. §103(a) should be reversed. Applicants respectfully request that this Honorable Board reverse the present rejections and direct that the aforementioned claims be issued.

Group 2 (claims 15-20 and 22)

Claims 15-20 and 22 all stand rejected under 35 U.S.C. § 103(a) as being unpatentable over *Yung* and *Luan* further in view of *Nishimoto* or, in the alternative simply over *Luan* in view of *Nishimoto* .

Nishimoto was cited by the Office simply to show a VLIW processor. *See* Official Action dated May 1, 2002, incorporated into later Official Actions dated October 18, 2002, April 9, 2003, and November 25, 2003. In an effort to focus issues for appeal, Applicant does not contest the use of *Nishimoto* as disclosure of a decoder for decoding a very long instruction word. Applicant reserves any issue regarding propriety of the combination; however, as explained below, this Appeal may be resolved based on proper interpretation of *Yung* and *Luan*.

Referring to independent claim 15, Applicant respectfully notes that neither *Yung* nor *Luan* discloses or suggests ***functional units each coupled to, and associated with, a respective register file segment*** of a register file, wherein ***each register file segment is implemented as an addressable array and partitionable into global and local registers***. In particular, *Yung* does not disclose or suggest a register file segment that is implemented as an addressable array and that is partitionable, but rather implements a hierarchical register architecture in which a fixed global register file (290) is shared amongst functional units and in which fixed and separate local buffers (241d, 242d ...) are defined for respective functional units to cache register values close to respective uses thereof. *See Yung*, Fig. 2, col. 6, lines 18-44. Global register file (290) and a separate local buffer (241d or 242d) do not together constitute a "register file segment" as

claimed and cannot be said to be associated with any respective functional unit. Indeed, no association exists between any global register and any particular functional unit in *Yung*.

Neither *Luan* nor *Nishimoto* provides the missing disclosure. Accordingly, for at least these reasons, all rejections (based on *Yung*) of the group 2 claims are unsustainable and should be reversed.

Luan is also devoid of disclosure of a register file segment that is implemented as an addressable array and that is partitionable. To the contrary, *Luan* discloses a:

> programmable shared memory system and method [that] selectively dedicates a first portion of memory to use by a processor and allocates a second portion of memory to shared use by the processor and any peripherals in the system. The programmable memory architecture is implemented using a dual bus architecture and a plurality of configurable memory banks. The dual bus architecture has a first bus connected to the processor and a second bus coupled to the peripherals by peripheral controllers and also coupled to the processor by a system controller.

Luan, col. 2, lines 3-13.

As before, to even attempt to interpret *Luan's* programmable shared memory as the same as (or interchangeable with) a register file segment that is implemented as an addressable array and that is partitionable, simply neglects the clearly distinct roles memory and registers play in computer system architecture. Computer systems consistent with Applicant's design, *Yung's* design or *Luan's* design, each demonstrably or necessarily and inherently employ both memory and registers. That memory and those registers are distinct, are employed for different purposes and are not the same or interchangeable. Therefore, whatever *Luan's* teaching with respect to selectively coupling memory banks to an appropriate bus to define first and second portions thereof, *Luan* does not disclose or suggest a register file segment that is implemented as an addressable array and that is partitionable. As before, *Nishimoto* does not provide the missing disclosure. For at least these reason, all rejections (based on *Luan*) of the group 2 claims are unsustainable and should be reversed.

However, even beyond the substantial disconnect between *Luan's* techniques Applicant's claimed register configurations, any attempt to map *Luan's* structures onto Applicant's claims, the Office must construe a group (i.e., two or more) of *Luan's* memory banks to be a register file

segment coupled to and associated with a respective functional unit and partitionable into global registers and local registers. No reasonable interpretation of *Luan* supports an association between any group of *Luan's* memory banks and any particular functional unit of *Luan's* processor. As before, *Nishimoto* does not provide the missing disclosure.

For at least the foregoing reasons, the Office's rejection of claims of group 2 over *Luan* is unsustainable and should be reversed.

Turning now to rejections based on *Yung* in view of *Luan*, Applicant respectfully notes that (as demonstrated above) neither *Yung* nor *Luan* discloses or suggests register file segments each coupled to, and associated with, a respective functional unit and each implemented as an addressable array and partitionable into global and local registers. Accordingly, no *prima facie* case of obviousness has been made out. As before, *Nishimoto* does not provide the missing disclosure. Therefore, for at least the foregoing reasons, the Office's rejection of claims of group 2 over *Yung* in view of *Luan* are not sustainable and should be reversed.

With regard to dependent claim 22, Applicant further notes that claim language requires that:

> partitioning of the register file is programmable so that the number N_G of global registers and number N_L of local registers is selectable and variable.

As the Office relies on Luan for programmable partitionability, Applicant notes that any combination of *Yung* nor *Luan* necessarily rests on the premise that programmable partitionability of *Luan's* memory subsystem may be imported into *Yung* to somehow provide programmable partitionability of storage provided by *Yung's* local buffers and global register file. As previously explained, registers and memory are not the same or interchangeable as employed by Applicant, *Yung* or *Luan*. Accordingly, whatever *Luan's* teaching with respect to memory banks, it is simply not applicable to registers. However, even more important, there is simply no way that *Luan's* techniques could be employed to cause an entry of *Yung's* local buffers to behave as a global register or to cause an entry of *Yung's* global register file to behave as a local buffer. As before, *Nishimoto* does not provide the missing disclosure.

Action Requested: For at least the reasons set forth above, the relied upon references fail to teach or suggest, alone or in combination, all elements of independent claims 15. Accordingly, all claims of Group 2 (claims 15-20 and 22) as well as those dependent therefrom (claim 21) are allowable and rejections under 35 U.S.C. §103(a) should be reversed. Applicants respectfully request that this Honorable Board reverse the present rejections and direct that the aforementioned claims be issued.

Group 3 (claims 8, 21 and 28)

Dependent claims 8, 21 and 28 all stand rejected under 35 U.S.C. § 103(a) as being unpatentable over various collections of references previously discussed: i.e., over *Luan* alone, over *Yung* in view of *Luan*, over *Luan* in view of *Nishimoto* , or over *Yung* and *Luan* further in view of *Nishimoto* . Applicants separately group dependent claims 8, 21 and 28 to emphasize that, in addition to the reasons previous given, the Office has not identified any teaching or suggestion in the any of the relied upon references that:

> register file segments are partitioned into N_G global and N_L local registers where N_G plus N_L is equal to N, the register file having N_G + (M * N_L) total registers available for the M functional units, the number of address bits for addressing the N_G + (M * N_L) total registers being equal to the number of bits B that are used to address $N = 2^B$ registers.

Since there is no disclosure in any of the relied upon references of register addressing that employs a set of address bits to uniformly address global and local registers, let alone any disclosure of use of the claimed number of address bits to address both global and local registers, Applicants respectfully submit that no *prima facie* case of obviousness has been made out with respect to the claims of group 3.

Action Requested: For at least the reasons set forth above, claims of Group 2 (claims 8, 21 and 28) are allowable and rejections under 35 U.S.C. §103(a) should be reversed. Applicants respectfully request that this Honorable Board reverse the present rejections and direct that the aforementioned claims be issued.

CONCLUSIONS

For at least the foregoing reasons, Appellants' claimed invention would not have been obvious under 35 U.S.C. §103(a) over the cited prior art. Accordingly, this Honorable Board is respectfully requested to reverse the rejection of claims 1-28 and direct this application to be issued.

Respectfully submitted,

David W. O'Brien, Reg. No. 40,107
Attorney for Applicant(s)
(512) 338-6314
(512) 338-6301 (fax)

CERTIFICATE OF MAILING OR TRANSMISSION

I hereby certify that, on the date shown below, this correspondence is being

☒ deposited with the US Postal Service with sufficient postage as first class mail, in an envelope addressed to Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

☐ facsimile transmitted to the US Patent and Trademark Office.

22 Aug 04
Date

EXPRESS MAIL LABEL: _____

APPENDIX OF CLAIMS INVOLVED IN THE APPEAL

1. A processor comprising:

a plurality of functional units; and

a register file that is divided into a plurality of register file segments, each coupled to and associated with respective ones of the plurality of functional units, the register file segments each implemented as an addressable array and partitionable into global registers and local registers, the global registers being accessible by the plurality of functional units, the local registers being accessible by the functional unit associated with the register file segment containing the local registers, wherein the number of global registers and the number of local registers are programmably configurable.

2. A processor according to Claim 1 wherein:

the processor is a Very Long Instruction Word (VLIW) processor.

3. A processor according to Claim 1 wherein:

the local registers and global registers are addressed using register addresses in an address space that is defined for a register file segment/ functional unit pair.

4. A processor according to Claim 1 wherein:

the register file is a multi-ported register file.

5. A processor according to Claim 1 wherein:

the local registers in a register file segment are addressed using register addresses in a local register range outside the global register range that are assigned within a single register file segment/ functional unit pair.

6. A processor according to Claim 1 wherein:

register addresses in the local register range are the same for the plurality of register file segment/ functional unit pairs and address registers locally within a register file segment/ functional unit pair.

7. A processor according to Claim 1 wherein:

the register file includes M of the register file segments, with each of the M register file segments having N physical registers, the register file segments having a reduced number of read and/or write ports in comparison to an undivided register file.

8. A processor according to Claim 7 wherein:

the register file segments are partitioned into N_G global and N_L local register files where N_G plus N_L is equal to N, the register file having $N_G + (M * N_L)$ total registers available for the M functional units, the number of address bits for addressing the $N_G + (M * N_L)$ total registers being equal to the number of bits B that are used to address $N = 2^B$ registers.

9. A processor according to Claim 8 wherein:

partitioning of the register file is programmable so that the number N_G of global registers and number N_L of local registers is selectable and variable.

10. A processor according to Claim 1 wherein the register file is a storage array structure having R read ports and W write ports comprising:

a plurality of storage array storages;

the storage array storages having a reduced number of read ports so that the total number of read ports for the plurality of storage array storages is R read ports; and

the storage array storages having W write ports.

11. A processor according to Claim 10 wherein:

the storage array structure is a multi-port structure; and

the plurality of storage array storages includes four storage array storages each having three read ports and five write ports.

12. A processor according to Claim 10 wherein:
the storage array structure is a multi-port structure; and
the plurality of storage array storages includes four storage array storages each having three read ports and four write ports.

13. A processor according to Claim 10 wherein:
the writes for the global registers are fully broadcast so that all of the storage array storages are held coherent.

14. A processor according to Claim 10 wherein:
storage array storages include storage cells having a plurality of word lines and a plurality of bit lines, the word lines being formed in one metal layer, the bits lines being formed in a second metal layer.

15. A processor comprising:
a decoder for decoding a very long instruction word including a plurality of sub instructions, the sub instructions being allocated into positions of the instruction word;
a register file coupled to the decoder and divided into a plurality of register file segments, each register file segment implemented as an addressable array and partitionable into global registers and local registers; and
a plurality of functional units, each coupled to and associated with respective ones of the register file segments, ones of the plurality of sub instructions being executable upon respective ones of the plurality of functional units, operating upon operands accessible to the register file segment associated with the respective functional unit, the global registers being accessible by the plurality of functional units, the local registers in each register file segment being accessible by the functional unit associated with the register file segment.

16. A processor according to Claim 15 wherein:
the local registers and global registers are addressed using register addresses in an address space that is defined for a register file segment/ functional unit pair.

17. A processor according to Claim 15 wherein:
the register file is a multi-ported register file.

18. A processor according to Claim 15 wherein:
the local registers in a register file segment are addressed using register addresses in a local register range outside the global register range that are assigned within a single register file segment/ functional unit pair.

19. A processor according to Claim 15 wherein:
register addresses in the local register range are the same for the plurality of register file segment/ functional unit pairs and address registers locally within a register file segment/ functional unit pair.

20. A processor according to Claim 15 wherein:
the register file includes M of the register file segments, with each of the M register file segments having N physical registers, the register file segments having a reduced number of read and/or write ports in comparison to an undivided register file.

21. A processor according to Claim 20 wherein:
the register file segments are partitioned into N_G global and N_L local register files where N_G plus N_L is equal to N, the register file having $N_G + (M * N_L)$ total registers available for the M functional units, the number of address bits for addressing the $N_G + (M * N_L)$ total registers being equal to the number of bits B that are used to address $N = 2^B$ registers.

22. A processor according to Claim 21 wherein:
partitioning of the register file is programmable so that the number N_G of global registers and number N_L of local registers is selectable and variable.

23. A method of operating a processor, the processor including a plurality of functional units and a register file divided into a plurality of register file segments, each coupled to and

associated with respective ones of the plurality of functional units, the register file segments each implemented as an addressable array, the method comprising:

partitioning the register file segments into global registers and local registers;

operating the plurality of functional units;

accessing the global registers by the plurality of functional units;

accessing the local registers by the functional unit associated with the register file segment including the local registers; and

programmably partitioning the register file so that the number of the global registers and the number of the local registers are selectable and variable.

24. A method according to Claim 23 further comprising:

addressing the local registers and global registers using register addresses in an address space that is defined for a register file segment/ functional unit pair.

25. A method according to Claim 23 further comprising:

addressing the local registers in a register file segment using register addresses in a local register range outside the global register range that are assigned within a single register file segment/ functional unit pair.

26. A method according to Claim 23 further comprising:

addressing the local register range the same for the plurality of register file segment/ functional unit pairs and address registers locally within a register file segment/ functional unit pair.

27. A method according to Claim 23, wherein the register file include M of the register file segments, with each of the M register file segments having N physical registers, the register file segments having a reduced number of read and/or write ports in comparison to an undivided register file.

28. A method according to Claim 27 further comprising:

partitioning the register file segments into N_G global and N_L local register files where N_G plus N_L is equal to N; and

operating the register file having $N_G + (M * N_L)$ total registers available for the M functional units, the number of address bits for addressing the $N_G + (M * N_L)$ total registers being equal to the number of bits B that are used to address $N = 2^B$ registers.